CLAIMS

What is claimed is:

1. A method comprising compressing data for communication in a terminal services environment by:

finding an index in a lookup table that matches an initial sequence in data, wherein:

the lookup table includes a plurality of entries, each said entry being discoverable utilizing a particular one of a plurality of said indices; and

each said entry references whether a corresponding said index is located in a history buffer, and if so, further references one or more locations of the corresponding said index in the history buffer; and

if the corresponding said entry of the matching index references a plurality of said locations:

for each said location, comparing a sequence at the location having the matching index with a sequence in the data, said sequence including the initial sequence;

deriving a matching sequence from the comparison based on at least one of a length and the location of the sequence at each said location; and

representing the matching sequence using a representation that includes the length and the location of the matching sequence in the history buffer.

2. A method as described in claim 1, further comprising:

finding one said index in the lookup table for each said sequence in the data;

forming compressed data that includes one or more said representations; and

streaming the compressed data.

3. A method as described in claim 1, wherein the corresponding said entry of the matching index references a hash chain which includes each said location of the matching index in the history buffer.

4. A method as described in claim 1, wherein the initial sequence and the index are each composed of at least two bytes.

5. A method as described in claim 1, further comprising:

forming compressed data that includes the representation; and

streaming the compressed data over a network, wherein the data is formatted as one or more packets.

6. A method as described in claim 1, further comprising encoding at least one of the length and the location of the representation using an encoding technique selected from the group consisting of:

Huffman encoding;

arithmetic encoding;

prefix encoding; and

Markov encoding.

7. A method as described in claim 1, wherein:

if the corresponding said entry of the matching index does not reference any said location, encoding the initial sequence by Huffman encoding;

if the corresponding said entry of the matching index references a single said location:

comparing a sequence at the single said location having the matching index with the sequence in the data;

deriving a matching sequence from the comparison based on at least one of a length and the location of the sequence at the single said location; and

representing the matching sequence using a representation that includes the length and the single said location of the matching sequence in the history buffer; and

when each said sequence of the data is represented or encoded, streaming the data having the encoding or the representation.

8. A method as described in claim 1, wherein the comparison to derive the matching sequence is performed utilizing one or more thresholds selected from the group consisting of:

a number of said locations having the matching index to be compared;

a size of a value that describes each said location having the matching index; and

a size of a value that describes a length of the sequence at each said location that matches the sequence in the data that includes the matching index.

9. A method as described in claim 1, further comprising employing a cost function to determine if the representation utilizes less memory when stored than the matching sequence, and if so, forming compressed data that includes the representation.

10. A method as described in claim 1, further comprising determining whether the location of the matching sequence matches one of a plurality of locations in a last recently used (LRU) table, wherein:

each said location in the LRU table has a corresponding said LRU representation;

each said location in the LRU table describes one of a plurality of last recently used locations of sequences in previously streamed data; and

if the location of the matching sequence is included in the LRU table, the location of the matching sequence is encoded with a corresponding said LRU representation from the LRU table.

11. One or more computer-readable media comprising computer-executable instructions that, when executed, perform the method as recited in claim 1.

12. A method comprising compressing data for communication in a terminal services environment by:

adding data to a history buffer;

updating a lookup table that references the history buffer to include the added data, wherein:

the lookup table includes a plurality of entries, each said entry being discoverable utilizing a particular one of a plurality of indices; and

each said entry references whether a corresponding said index is located in a history buffer, and if so, further references one or more locations of the corresponding said index in the history buffer;

starting a current pointer at the added data in the history buffer;

finding one said index in the lookup table that matches an initial sequence at the current pointer;

if the corresponding said entry of the matching index references a plurality of said locations:

comparing a sequence at each said location having the matching index with a sequence in the added input data that includes the initial sequence;

deriving a matching sequence from the comparison;

representing the matching sequence with a representation that includes the location and a length of the matching sequence in the history buffer;

employing a cost function to determine if the representation utilizes less memory space when stored than the matching sequence,

if so, configuring data to include the representation and advancing the current pointer by the length of the matching sequence,

otherwise, configuring data to include the initial sequence and advancing the current pointer by a length of the initial sequence; and

when the current pointer has advanced through the added data, packetizing the configured data for streaming.

13. A method as described in claim 12, further comprising encoding at least one of the length and the location of the representation using an encoding technique selected from the group consisting of:

Huffman encoding;

arithmetic encoding;

prefix encoding; and

Markov encoding.

14. A method as described in claim 12, wherein when the initial sequence at the current pointer does not match any sequence in the history buffer, the initial byte sequence at the current pointer is encoded for inclusion into configured data for streaming by an encoding technique selected from the group consisting of:

Huffman encoding;

arithmetic encoding;

prefix encoding; and

Markov encoding.

15. A method as described in claim 12, further comprising determining whether the location of the matching sequence matches one of a plurality of locations in a last recently used (LRU) table, wherein:

each said location in the LRU table has a corresponding said LRU representation;

each said location in the LRU table describes one of a plurality of last recently used locations of sequences in previously streamed data; and

if the location of the matching sequence is included in the LRU table, the location of the matching sequence is encoded with a corresponding said LRU representation from the LRU table.

16. One or more computer-readable media comprising computer-executable instructions that, when executed, perform the method as recited in claim 12.

17. A method comprising:

receiving feedback that indicates availability of resources for communicating data over a network from a terminal service provided by a server to a client; and

tuning one or more parameters of a compression routine utilized to compress the data in response to the feedback.

18. A method as described in claim 17, wherein the resources are selected from the group consisting of:

hardware resources of the client;

software resources of the client;

network resources of the network;

hardware resources of the server;

software resources of the server; and

any combination thereof.

19. One or more computer-readable media comprising computer-executable instructions that, when executed, perform the method as recited in claim 17.

20. A method comprising:

receiving a request at a server from a client for remote access to an application or a file that is available through the server;

determining availability of resources for communicating data in response to the request over a network; and

tuning one or more parameters of a compression protocol utilized to compress the data based on the determined availability.

21. A method as described in claim 20, wherein the resources are selected from the group consisting of:

hardware resources of the client;

software resources of the client;

network resources of the network;

hardware resources of the server;

software resources of the server; and

any combination thereof.

22. One or more computer-readable media comprising computer-executable instructions that, when executed, perform the method as recited in claim 20.

23. A server comprising:

a history buffer having a plurality of bytes;

a lookup table that includes a plurality of entries, each said entry:

being discoverable utilizing a particular one of a plurality of indices; and

references whether a corresponding said index is located in the history buffer, and if so, further references one or more locations of the corresponding said index in the history buffer; and

a compression module that is executable to:

find one said index sequence in the lookup table that matches an initial sequence in data for communication to a client from a terminal service;

if the corresponding said entry of the matching index references a plurality of said locations:

for each said location, compare a sequence at the location having the matching index with a sequence in the data, said sequence including the initial sequence;

derive a matching sequence from the comparison based on at least one of a length and the location of the sequence at each said location; and

represent the matching sequence using a representation that includes the length and the location of the matching sequence in the history buffer.

24. A server as described in claim 23, wherein the compression module is

further executable to:

find one said index in the lookup table for each said sequence in the data;

form compressed data that includes one or more said representations; and

stream the compressed data;

25. A server as described in claim 23, wherein the corresponding said entry of the matching index references a hash chain which includes each said location of the matching index in the history buffer.

26. A server as described in claim 23, wherein the initial sequence and the index are each composed of at least two bytes.

27. A server as described in claim 23, wherein the compression module is further executable to:

form compressed data that includes the representation; and

packetize the compressed data for streaming over a network, wherein the data is formatted as one or more packets.

28. A server as described in claim 23, wherein the compression module is further executable to encode at least one of the length and the location of the representation by an encoding technique selected from the group consisting of:

Huffman encoding;

arithmetic encoding;

prefix encoding; and

Markov encoding.

29. A server as described in claim 23, wherein the compression module is further executable to encode the initial sequence if the corresponding said entry of the matching index does not reference any said location by an encoding technique selected from the group consisting of:

Huffman encoding;

arithmetic encoding;

prefix encoding; and

Markov encoding.

30. A server as described in claim 23, wherein the comparison to derive the matching sequence is performed utilizing one or more thresholds selected from the group consisting of:

a number of said locations having the matching index to be compared;

a size of a value that describes each said location having the matching index; and

a size of a value that describes a length of the sequence at each said location that matches the sequence in the data that includes the matching index.

31. A server as described in claim 23, wherein the compression module is further executable to employ a cost function to determine if the representation utilizes less memory when stored than the matching sequence, and if so, forming compressed

data that includes the representation.

32. A server as described in claim 23, wherein the compression module is further executable to determine whether the location of the matching sequence matches one of a plurality of locations in a last recently used (LRU) table, wherein:

each said location in the LRU table has a corresponding said LRU representation;

each said location in the LRU table describes one of a plurality of last recently used locations of sequences in previously streamed data; and

if the location of the matching sequence is included in the LRU table, the location of the matching sequence is encoded with a corresponding said LRU representation from the LRU table.

33. A system comprising:

a network;

a server including:

a first history buffer having a plurality of bytes;

a lookup table that includes a plurality of entries, each said entry being discoverable utilizing a particular one of a plurality of indices, each said entry references whether a corresponding said index is location in the history buffer, and if so, one or more locations of the corresponding said index in the history buffer; and

a compression module that is executable to:

find one said index in the lookup table that matches an initial sequence at a current pointer in data to be streamed in response to a request for remote access;

if the corresponding said entry of the matching index references one or more said locations:

compare a sequence at each said location having the matching index with a sequence in the data at the current pointer;

derive a matching sequence from the comparison;

configure data to include a representation that includes the location and a length of the matching sequence in the first history buffer and advance the current pointer by the length of the matching sequence;

if the corresponding said entry of the matching index does not reference any said location, configure data to include the initial sequence and advancing the current pointer by a length of the initial sequence; and

when the current pointer has advanced through the added data, stream the configured data over the network; and

a client communicatively coupled to the network and including a second said history buffer and a decompression module that is executable to decompress the streamed data by finding the matching sequence in the second said history buffer based on the location and the length indicated by the representation.

34. A system as described in claim 33, wherein the decompression module is

further executable by the client to add decompressed data to the second history buffer.

35. A system as described in claim 33, wherein the compression module is further executable by the server to encode at least one of the length and the location of the representation by an encoding technique selected from the group consisting of:

Huffman encoding;

arithmetic encoding;

prefix encoding; and

Markov encoding..

36. A system as described in claim 33, wherein if the corresponding said entry of the matching index references does not reference any said location, the compression module is further executable to encode the initial sequence by an encoding technique selected from the group consisting of:

Huffman encoding;

arithmetic encoding;

prefix encoding; and

Markov encoding..

37. A system as described in claim 33, wherein the compression module is further executable by the server to determine whether the location of the matching sequence matches one of a plurality of locations in a last recently used (LRU) table, wherein:

each said location in the LRU table has a corresponding said LRU representation;

each said location in the LRU table describes one of a plurality of last recently used locations of sequences in previously streamed data; and

if the location of the matching sequence is included in the LRU table, the location of the matching sequence is encoded with a corresponding said LRU representation from the LRU table.

38. A computer-readable medium comprising computer-executable instructions that, when executed by a computer, direct the computer to:

find an index in a lookup table that matches an initial sequence in data for streaming to a client, the data for generating a user interface of an application that is being executed remotely from the client, wherein:

the lookup table includes a plurality of entries, each said entry being discoverable utilizing a particular one of a plurality of said indices; and

each said entry references whether a corresponding said index is located in a history buffer; and if so, further references one or more locations of the corresponding said index in the history buffer;

if the corresponding said entry of the matching index references a plurality of said locations:

for each said location, compare a sequence at the location having the matching index with a sequence in the data, said sequence including the initial sequence; and

compute, from the comparison, a length of the matching sequence.

39. A computer-readable medium as described in claim 38, wherein the computer-executable instructions direct the computer to represent the matching sequence using a representation that includes the length and the location.

40. A computer-readable medium as described in claim 38, wherein the corresponding said entry of the matching index references a hash chain which includes each said location of the matching index in the history buffer.

41. A computer-readable medium as described in claim 38, wherein the initial sequence and the index are each composed of at least two bytes.

42. A computer-readable medium as described in claim 38, wherein the computer-executable instructions direct the computer to encode at least one of the length and the location by an encoding technique selected from the group consisting of:

Huffman encoding;

arithmetic encoding;

prefix encoding; and

Markov encoding.

43. A computer-readable medium as described in claim 38, wherein the computer-executable instructions direct the computer to encode the initial sequence if the

corresponding said entry of the matching index does not reference any said location by an encoding technique selected from the group consisting of:

Huffman encoding;

arithmetic encoding;

prefix encoding; and

Markov encoding.

44. A computer-readable medium as described in claim 38, wherein the comparison is performed utilizing one or more thresholds selected from the group consisting of:

a number of said locations having the matching index to be compared;

a size of a value that describes each said location having the matching index; and

a size of a value that describes a length of the sequence at each said location that matches the sequence in the data that includes the matching index.

45. A computer-readable medium as described in claim 38, wherein the computer-executable instructions direct the computer to employ a cost function to determine if a representation that includes the length and the location of the matching sequence utilizes less memory when stored than the matching sequence, and if so, form compressed data that includes the representation.

46. A computer-readable medium as described in claim 38, wherein the computer-executable instructions direct the computer to determine whether the location

of the matching sequence matches one of a plurality of locations in a last recently used (LRU) table, wherein:

each said location in the LRU table has a corresponding said LRU representation;

each said location in the LRU table describes one of a plurality of last recently used locations of sequences in previously streamed data; and

if the location of the matching sequence is included in the LRU table, the location of the matching sequence is encoded with a corresponding said LRU representation from the LRU table.